IOSR Journal of Engineering (IOSR JEN) **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
PP 34-37

Recipe Recommender

Riddhi Mayekar[#*1], Siddhi Kamat[#2], Saish Banaulikar[#3], Hrushikesh Sawant[#4], Shannon Vaz[#5], Basil Jose[#6]

[#]*Computer Department*
Agnel Institute of Technology and Design Mapusa, India

Abstract: *It is difficult to think of recipe to prepare a new dish based on ingredients available at hand. Humans tend to keep a record of recipes but this set is limited to the recipes a person has encountered before. Each recipe has a set of instructions for preparing a particular dish, including list of ingredients.*
This application takes ingredients to be included as well as ingredients to be excluded as specified by user and retrieves list of recipes that match the given input. An alternative way is to search for a recipe based on the recipe name.
This recommender system is a content based filtering system in which the system uses previously searched data and accordingly recommends recipes which the user might be interested in. The recipes shown to the user would be based on the user's preference, which is specified by the user when he or she creates an account. It will also be based on search history of the user. For unregistered users, the recipes retrieved will be based on the recipes currently viewed and the subset of ingredients in that recipe.
Keywords*: recipe, recommend, ingredient, application, search engine*

I. Introduction

The recent trend is such that people rely on the internet for any information such as shopping for a dress to cat videos. There is too much information and the user is not interested in all of it. This applies to many fields like clothing, food etc. Hence there is a need to filter information according to user preference. So that the user can retrieve the data in a short amount of time.

A recommender system is an information filtering system that provides personal preference guide based on the user profile and preferences. There are two approaches to implement recommender systems :

- Collaborative Filtering : Recommendations which directly utilize the user's feedback.
- Content Filtering : Recommendations which use the core content.

A. Problem statement

For a layman, it is difficult to think of a recipe based on the ingredients which are present with him and decide which recipe to prepare.

There are many sources of recipes on the web but the user may not be interested in all of them. We need a recommendation system which will filter out this information and recommend a list of recipes based on user preference. This will allow users to make better and quick decision to choose a recipe.

B. Motivation

The amount of information is ever increasing. With the advent of mass data sharing technologies like the world wide web, information is readily available anytime and can be accessed by anyone on mobile devices with an Internet connection. But all the information available is not what the user is actually interested in, hence we need to filter it out and present to the user only the information he/she is interested in.

The human mind has a limited capacity to remember and recollect information. Humans also tend to spend a lot of time taking decisions like for what to cook for dinner. People's lives are getting busier and they do not want to waste their time taking trivial decisions like deciding what to cook for lunch. Hence the availability of a system to suggest what to cook would greatly relieve us from this dilemma.

Information retrieval and information filtering techniques are aimed at solving the issue of presenting only relevant information to the user. A sub class of information filtering systems are recommender systems which aim at predicting what information the user would be interested in.

C. Scope

The scope of the proposed system is to develop a recommender system for recipes which would suggest users what recipe they should try to cook. The system uses data mining approach to make predictions based on what the user has viewed already to suggest recipes to the user.

D. Recommendation System [1]

A recommendation system is a software which, with the help of recommender algorithms, tries to predict what information the user would need based on their interest. The interests of the user are captured in various ways like history of viewed content, ratings given by user to content, user reviews and so on.

Based on what method is used for capturing user interests and how the predictions are made, the recommendation algorithms can be categorized as follows:

Content based recommendation systems : These type of recommendation systems use the description of the content to find similar content from the available information and recommend it. For example in case of recipes, the content would be the ingredients used, the preparation time, cuisine etc.

Collaborative recommendation systems : These type of recommendation systems use feedback from multiple users like ratings and reviews of users to give relevance to a particular item for a user. For example if many users have highly rated an item then a user would be recommended that item.

The system proposed is based on content-based approach which will produce suggestions by examining recipes with similar content.

II. Design

A. Flow Diagram

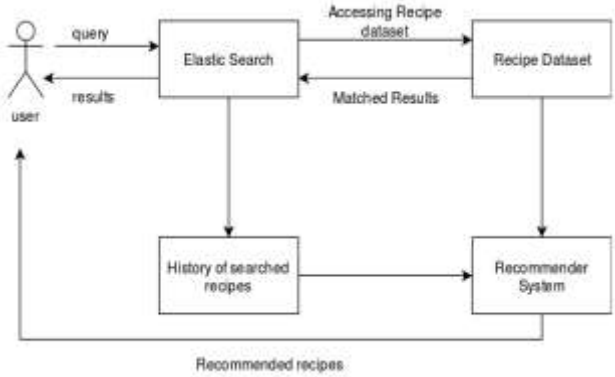

Fig : Flow Diagram Of Overall System

User inputs list of ingredients to be included and excluded or recipe name in the form of a query. This query is given to the elastic search engine as input, which processes this query and access the data from recipe dataset. It retrieves recipes which matches with the query and displays it to the user.

B. Topic Modeling

Topic modeling is a type of statistical modeling for discovering the abstract "topics" that occur in a collection of documents. Topic models provide a simple way to analyze large volumes of unlabeled data. Topic consists of a cluster of words that occur together in the similar context. The model organizes the collection of data according to the topics discovered[3].

C. Data Preprocessing

The input is the JSON format which is preprocessed before giving it to topic modeling algorithm.The steps forpreprocessing are as follows:

- Each recipe has few metrics which need to be converted to a standard metric. In our case we have converted measures like cups, teaspoons and tablespoon to grams.
- Once converted into grams, the values are divided by serving size of that recipe.
- Then using the above value in MinMax formula we get a certain value for each ingredient.
- We find the No. of words for a particular ingredient by diving the MinMax value for that ingredient by total MinMax value.
- Now on the basis of No. Of words we create the bag of words.

D. Topic modeling Algorithm

Latent Dirichlet Allocation (LDA) is an example of topic model and is used to classify text in a document to a particular topic. It builds a topic per document model and words per topic model. It takes the whole set of documents and splits them into topics discovered.

III. Proposed System

Our dataset is downloaded from yummly.com which is one of the best recipe sharing website. Each recipe containing list of ingredients and steps taken to prepare the dish. We have downloaded about 9,570 recipes from yummly for our system.

A. Latent Dirichlet Allocation

LDA is a type of topic modeling algorithm. It is a probabilistic model for discovering topics that are present in documents. Documents are represented as a probability distribution over topics and topics are probability distribution over words. The documents are considered similar if they have similar topic distribution as the topics will tend use similar words[2].

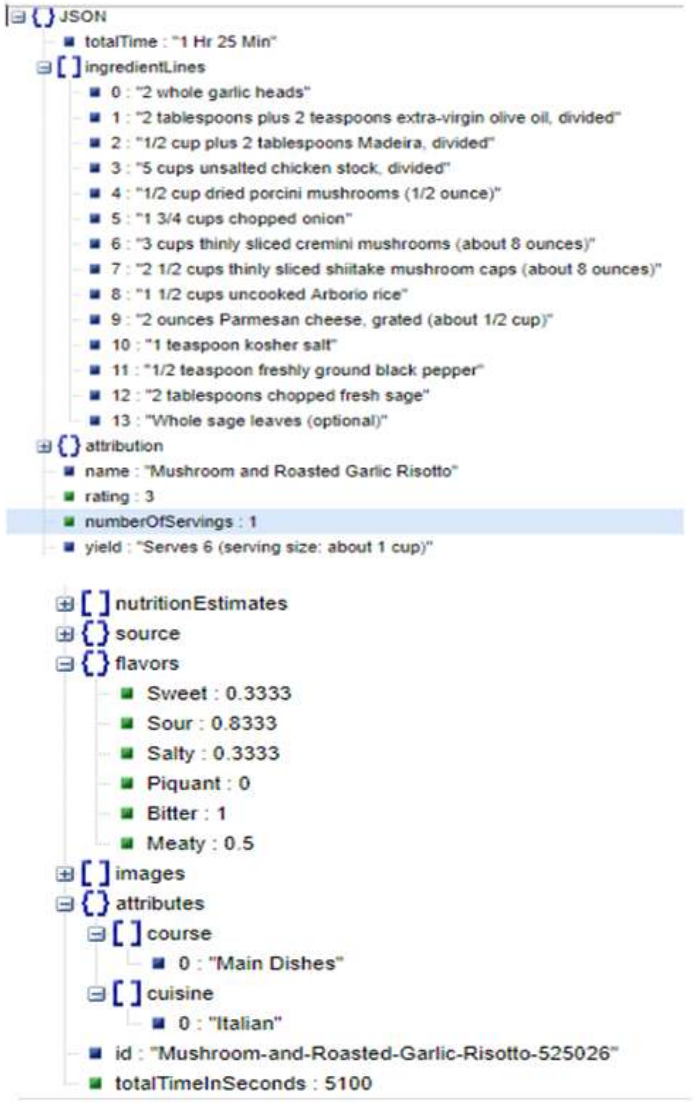

Fig: Dataset in JSON format

LDA uses the following 3 steps :

Step 1: User needs to input number of topics to the algorithm. It can be done either by using previous analysis or simply by trial-and-error.

Step 2: The algorithm will assign every word to a temporary topic. Initially topic assignments will be done on the basis of tf-idf score. It will be temporary as it will be further updated in step 3.

Step 3: The algorithm will check and update the topic assignments.

For each word in every document, it updates topic assignment by carrying out the following calculations:

1. Proportion of topics in the document
X=p(topic t | document d) = proportion of words in document d that are currently assigned to topic t.

2. Proportion of word across topics
Y=p(word w | topic t) = proportion of assignments to topic t for word w over all the documents[4].

B. Jensen-Shannon Divergence [6]

After applying LDA on the documents we obtain the distribution of topics in the documents. Now inorder to recommend the recipes we need to find similar recipes to one the user is currently viewing. This can be done by finding the measures of divergence between the two recipes.

Jensen Shannon Divergence is the symmetric measure which provides the better measure of divergence than the KullbackLeibler Divergence. It always the value between 0 and 1 inclusive of both. If the Jensen-Shannon divergence value is tending to 0 that means the distribution are similar else they are dissimilar.

C. Elasticsearch [5]

Elasticsearch is a near real time search which means documents are available quickly after it has been indexed. It is highly scalable, open-source, full-text search and analytics engine which allows one to store, search and analyze big volume of data very efficiently and in real time.

Elasticsearch is a standalone database server which is written in java. It takes data and stores it in sophisticated format. It optimizes the date inputted for language based searches, which makes working with it convenient. Since main protocol is implemented using HTTP/JSON.

Application Programming Interface/API is a group of function calls or other programming instructions to access the software component in that particular application. Elasticsearch provides REST API that can be accessed by JSON over HTTP.

D. Advantages Of Elasticsearch

- It stores complex entities as structured JSON documents and index all fields by default which leads to the higher performance.
- Since it is built on top of Lucene it offers full-text search capabilities which is most powerful function of it.
- It is schema free which means large quantity of JSON data is stored in distributed fashion and also indexes the data to make it search friendly.
- It performs linguistic search against the document and returns the matched document to the condition specified. Scalable upto petabytes of structured and unstructured data.

IV. Conclusion

The system proposed is a content based recommender system which recommends recipes to users. The system records the preferences of the user based on history of viewed recipes and uses this information to make recommendations.

The recommender system uses topic modeling approach to identify the underlying thematic structure of a corpus of recipes and generates a distribution of topics in documents and a distribution of words in topics. This is done using LDA, a generative probabilistic model for discovering topics in a corpus.

We begin by first extracting the recipes from the web (yummly.com) in JSON format. These recipes are processed and converted to bag-of-words style documents based on the amount of ingredient that goes into the recipe. These documents along with the number of expected topics is inputted to the algorithm (LDA). LDA will then output the topics distribution in the corpus.

Users will run queries in the software based on the recipe name/ingredients available. The query will be processed by Elasticsearch search engine and showed to the user. The queries given will be recorded to produce recommendations. The divergence is found between the recipes retrieved previously and the remaining recipes. Recipes with least divergence are recommended to the user.

References

[1]. Jagithyala, A., 2014. Recommending recipes based on ingredients and user reviews (Doctoral dissertation, Kansas State University).
[2]. Blei, D.M., 2012. Probabilistic topic models. Communications of the ACM, 55(4), pp.77-84.
[3]. Blei, D.M., Ng, A.Y. and Jordan, M.I., 2003. Latent dirichlet allocation. Journal of machine Learning research, 3(Jan), pp.993-1022.
[4]. Barhate, P. (2018). Latent Dirichlet Allocation for Beginners: A high level intuition. [online] Medium. Available at: https://medium.com/@pratikbarhate/latent-dirichlet-allocation-for-beginners-a-high-level-intuition-23f8a5cbad71
[5]. Elastic.co. (2018). Elasticsearch: RESTful, Distributed Search & Analytics | Elastic. [online] Available at: https://www.elastic.co/products/elasticsearch
[6]. Lin, J., 1991. Divergence measures based on the Shannon entropy. IEEE Transactions on Information theory, 37(1), pp.145-